Takayuki SHIMATANI, Application No. 10/578,232   Dkt. 2271/76217
Page 7

Amendments to the Drawings

The replacement sheet of drawings attached hereto as **Exhibit A** include changes to Fig. 2, to correct an unintentional error in the amended Fig. 2 submitted on November 4, 2009 which inadvertently showed an arrow from two-way communication module 21 to port monitor 13.

Fig. 2 has been amended to show the arrow from two-way communication module 21 of the client PC 2 to language monitor 12 of the host PC 1.

Attachment: replacement sheet of drawings for Fig. 2

Takayuki SHIMATANI, Application No. 10/578,232   Dkt. 2271/76217
Page 8

REMARKS

Claims 1-10 are pending. By this Amendment, claim 1 has been amended to correct an informality therein. Entry of the Amendment is requested. Claims 1-10 remain pending upon entry of this Amendment, with claims 1, 3 and 6 being in independent form.

The drawings were objected to as having informalities. The replacement sheet for Fig. 2 attached hereto as **Exhibit A** corrects an unintentional error in the amended Fig. 2 submitted on November 4, 2009 which inadvertently showed an arrow from two-way communication module 21 to port monitor 13. Fig. 2 has been amended to show the arrow from two-way communication module 21 to language monitor 12. Applicant submits that no new matter is introduced by the drawing changes.

Claims 1, 3, 4, 6 and 7 were rejected under 35 U.S.C. § 103(a) as purportedly unpatentable over Kadota (US 2004/0034862 A1) in view of Tokura (US 6,879,410). Claims 2, 5 and 8-10 were rejected under 35 U.S.C. § 103(a) as purportedly unpatentable over Kadota in view of Tokura and further in view of Ohta (US 2006/0001908 A1).

Applicant respectfully submits that the present application is allowable over the cited art, for at least the reason that the cited art does not disclose or suggest the aspects of the present application of (a) obtaining from a registry of the client computer a two-way communication flag which is set by a user utilizing a user interface of the client computer, and if the two-way communication flag is in the state where the two-way communication is enabled, enabling two-way communication regardless of an operating system of the client computer and obtaining status information of the printer via the server computer, (b) obtaining from the registry, in a case where the two-way communication is enabled, printer information set by the operating system of the client computer, and (c) determining whether a two-way flag of the printer information is ON
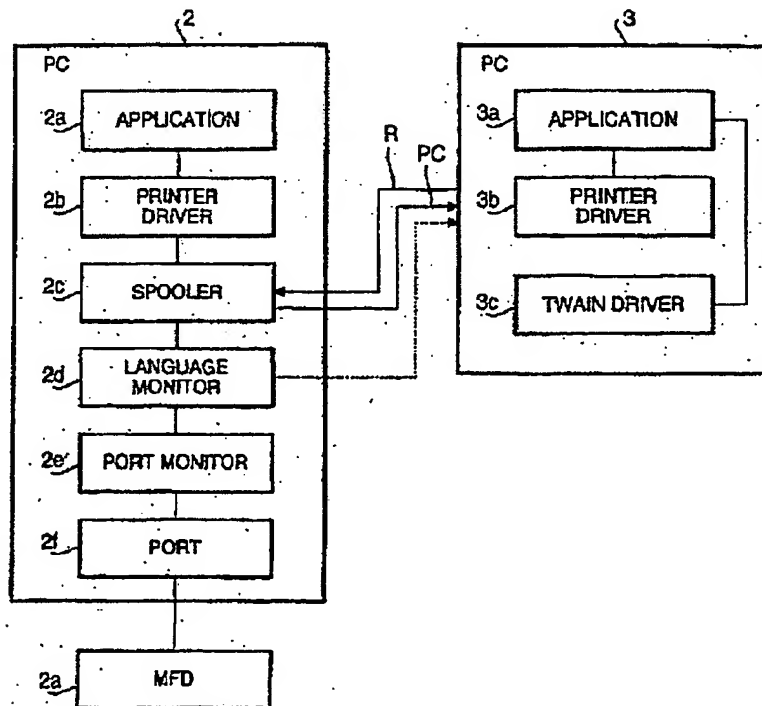
Takayuki SHIMATANI, Application No. 10/578,232   Dkt. 2271/76217
Page 9

after the printer information is obtained, and in a case where the two-way flag is ON, executing

the two-way communication.

Kadota, as understood by applicant, proposes a configuration of a printing system 100, as

shown in Fig. 3 (reproduced below) of Kadota, wherein an MFP (Multi-Function Peripheral) 1 is

connected to a PC 2, which in turn is connected to a PC 3. The PC 2 includes printer driver 2b

(for the printer function of the MFP 1), spooler 2c (which temporarily spools print data), a

language monitor 2d (which checks progression of print operation), port monitor 2e (which

designates a port from which the print data is output), and port 2f to which the MFP 1 is

connected. The PC 3 has includes printer driver 3b (for the printing function of the MFP 1) and

TWAIN driver 3c (driver software for the scanner function of the MFP 1).

FIG. 3

Takayuki SHIMATANI, Application No. 10/578,232   Dkt. 2271/76217
Page 10

Kadota, [0107]-[0111] (reproduced below), was cited in the Office Action:

[0107] If the operating system *supports the bi-directional communication between the spooler and the printer function* (e.g., Windows 98®), with use of the API (application program interface), which is provided by the printer driver 2b or the printer driver 3b, for controlling the printer function (when the operating system is Windows®, ReadPrinter which is the API for reading data from the printer), TWAIN driver in the PC 2 or PC 3, which transmitted the print command, can obtain data from the designated printer (i.e., MFP 1) through the spooler 2c, using the RPC, as indicated by the solid line in FIG. 3.

[0108] If the operating system *does not support the bi-directional communication between the spooler and the MFP* (e.g., Windows NT®), as indicated by broken lines in FIG. 3, using a Named Pipe. TWAIN driver in the PC 2 or PC 3 can obtain data from the designated printer (i.e., MFP 1) through the language monitor 2d.

[0109] It should be noted that the *Named Pipe is generally used for enabling a communication between applications, and is provided by the operating system* (e.g. Windows NT®). *With use of the Named Pipe, a communication between different PCs is also possible*. This architecture is similar to a method for transmitting data from the language monitor to the status monitor provided to the printer driver, which has been conventionally done and will not be described in detail.

[0110] In the first embodiment, a TWAIN driver 3c, which is the driver software of the scanner function of the MFP 1, includes a program for controlling the MFP 1. Then, with use of the configuration of the printing system described above, a read command for the scanner function of the MFP 1 issued by the application 3a is transmitted to the PC 2 by the RPC (Remote Procedure Call) command using the program interface for the printer function (i.e. when the operating system is Windows®, the WritePrinter command which is the API for the printer). The MFP 1 makes determination by analyzing data/command transmitted by each drive.

[0111] Similarly, when the scanned data is received, the TWAIN driver 3c receives the data, using the program interface for the printer function (i.e., when the operating system is Windows®, the ReadPrinter command which is the API for the printer), through the PC 2, using the RPC or the Named Pipe.

Thus, Kadota proposes various approaches for communication between the spooler and

the printer function (for example in multi-function device, MFP), depending on whether the

operating system supports bi-directional communication between the spooler and the printer

function, and proposes use of a Named Pipe or RPC (remote procedure call) for communication

between PCs. In addition, Kadota proposes use of a TWAIN driver for the PC to communicate

Takayuki SHIMATANI, Application No. 10/578,232   Dkt. 2271/76217
Page 11

with the scanner function of the MFP (multi-function device).

Kadota, [0151]-[0156] (reproduced below), which was also cited in the Office Action,

sets forth details regarding a proposed operations sequence for scan operations:

> [0151] Next, an example of operation sequence of the PC3 and MFP 1 will be described with reference to FIG. 12.
>
> [0152] When the *scan start command* is issued by the application in the PC 3 (S1301), the TWAIN driver 3c of the PC 3 calls the WritePrinter (API for the printer), which is transmitted to the MFP 1 as the scan mode moving command (S1303). At this stage, in this example, the MFP 1 operates in the idle mode (S1302). The MFP 1 changes it operation mode to the scan mode upon receipt of the scan mode moving command (S1304).
>
> [0153] The TWAIN driver of the PC 3 calls the ReadPrinter (API for the printer) in order to obtain the ACK signal from the MFP 1 (S1305). Next, the MFP 1 transmits the ACK signal to the PC 3 through the PC 2 (S1306). When the ACK is received, the PC 3 operates to accept the scan setting input by the user (S1307).
>
> [0154] When the TWAIN driver 3c of the PC 3 accepts the *scan setting* input, the TWAIN driver 3c calls the WritePrinter, which is the API for the printer as the command for scan setting and starting and transmits the command to the MFP 1 through the PC 2 (S1308). When the MFP 1 receives the scan setting and starting command is received, the MFP 1 changes the settings in accordance with the received data (S1309), and executes the scanning operation (S1310).
>
> [0155] The TWAIN driver 3c of the PC 3 calls the ReadPrinter, which is the API for the printer, in order to receive the scan data from the MFP 1 through the PC 2, and waits for the response from the MFP 1 (S1311). In response to the ReadPrinter (API), the MFP 1 transmits the scan data to the PC 3 through the PC 2 (S1312). The steps S1311 and S1312 are repeated until the data for one page of the original is scanned and transmitted.
>
> [0156] After the scanning for one page is finished (S1313), the MFP 1 transmits the sheet end status (S1314), and changes its operation mode to the idle mode (S1315).

Although Kadota proposes that the TWAIN driver for the scanner function of the MFP

can be used to control the MFP in terms of *scan settings*, Kadota, as acknowledged in the Office

Action, says nothing whatsoever regarding a user interface through which a user can set a two-

way communication flag stored in a registry of the client computer.

Further, at no point does Kadota teach or suggest obtaining from a registry of the client

computer, *in a case where the two-way communication is enabled*, printer information set by the operating system of the client computer.
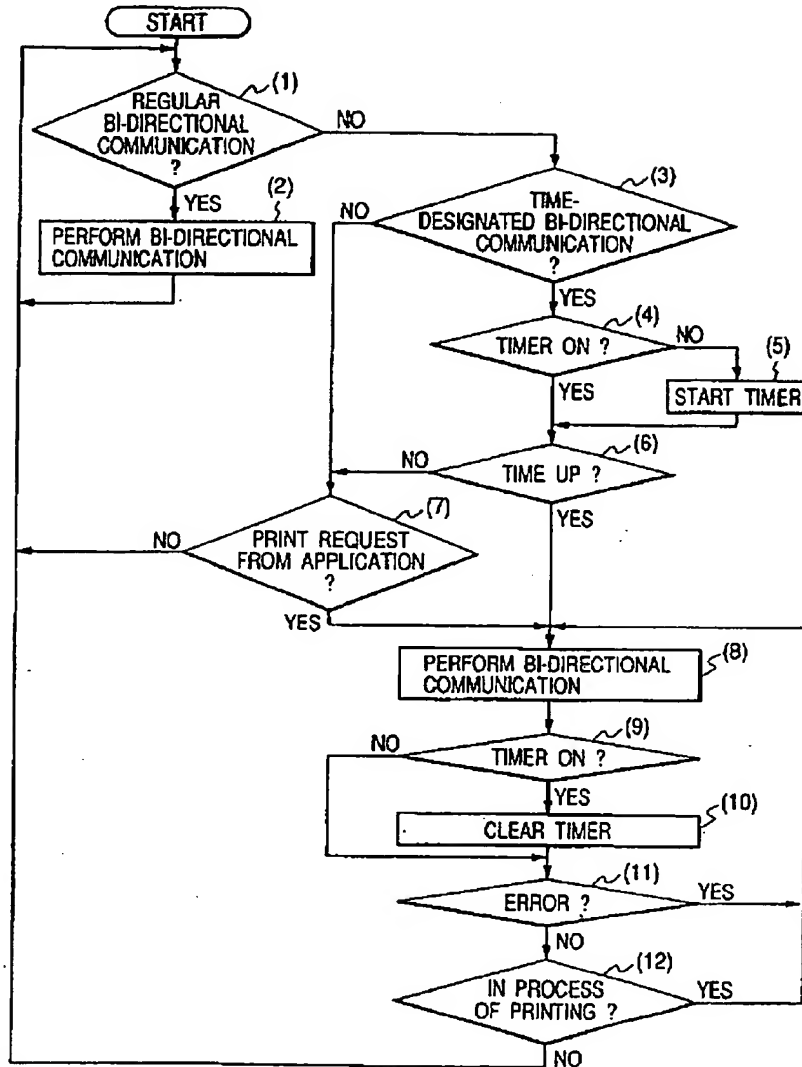
Should the Examiner disagree therewith, the Examiner is requested to indicate by element, paragraph and line numbers the portions of Kadota equated with (i) a registery of the client computer, (ii) a case where the two-way communication is enabled, and (iii) printer information set by the operating system of the client computer.

Tokura, as understood by applicant, proposes a data processing apparatus configured, as shown in Fig. 2 (reproduced below), to prevent or reduce interference (cause by continuous bi-directional communication between the host and a printer) to the operation of a power saving function of a host device. Tokura assumes that bi-directional communication must be performed either continuously or intermittently, in order to detect various states of the printer, and in the approach proposed by Tokura, the user can select an intermittent bi-directional communication mode wherein the bi-directional communication is allowed for obtaining printer status information between the host and the printer only at certain time intervals, giving priority to the power saving function mode (step (3), "YES"; Tokura, col. 1, lines 57-61). In such approach, the user can freely select a print request accompanied with a desired power saving function and/or a print request which is not accompanied with the power saving function (Tokura, col. 5, lines 62-65).

FIG. 2

```
                    ┌──────────┐
                    │  START   │
                    └──────────┘
                         │
                       ╱(1)╲
              REGULAR        NO
           BI-DIRECTIONAL ──────────────────────┐
            COMMUNICATION                        │
                 ?                            ╱(3)╲
                 │ YES        (2)      TIME-
                 ▼       NO  DESIGNATED BI-DIRECTIONAL
        ┌──────────────┐◄── COMMUNICATION
        │ PERFORM BI-DIRECTIONAL │        ?
        │ COMMUNICATION │      │ YES
        └──────────────┘     ╱(4)╲    NO      (5)
                        TIMER ON ? ──────► START TIMER
                             │ YES
                           ╱(6)╲
              NO      TIME UP ?
          ◄───────
                           │ YES
         ╱(7)╲
  PRINT REQUEST    NO
  FROM APPLICATION ◄───
       ?
       │ YES
       ▼
┌──────────────────────┐  (8)
│ PERFORM BI-DIRECTIONAL │
│ COMMUNICATION          │
└──────────────────────┘
       │
     ╱(9)╲
  NO TIMER ON ?
       │ YES       (10)
  CLEAR TIMER
       │
     ╱(11)╲       YES
    ERROR ? ──────►
       │ NO
     ╱(12)╲      YES
  IN PROCESS ──────►
  OF PRINTING ?
       │ NO
```

While the user can specify a time interval between intermittent bi-directional communications, Tokura, contrary to the contention in the Office Action, does not disclose or suggest (a) obtaining from a registry of the client computer a two-way communication flag which is set by a user utilizing a user interface of the client computer, and if the two-way

communication flag is in the state where the two-way communication is enabled, enabling two-way communication regardless of an operating system of the client computer and obtaining status information of the printer via the server computer, (b) obtaining from the registry, in a case where the two-way communication is enabled, printer information set by the operating system of the client computer, and (c) determining whether a two-way flag of the printer information is ON after the printer information is obtained, and in a case where the two-way flag is ON, executing the two-way communication.

As mentioned above, merely proposes that the timing of two-way communication can be controlled according to timing designations by the user.

The approach of Tokura simply does not involve, and Tokura does not teach or suggest, disabling two-way communication if the two-way communication flag is set to a corresponding value. While the two-way communication for obtaining printer status information between the host and the printer is controlled depending on whether continuous or intermittent mode is selected by the user, nowhere in Tokura is disclosed or suggested the aforementioned aspect of setting a two-way communication flag in the registry of the client computer by a user utilizing a user interface, nor obtaing such two-way communication flag setting from said registry.

As indicated in Tokura, column 6, lines 1-14, even when the apparatus of Tokura is in intermittent bi-directional communication mode (not always performing two-way communication), the apparatus remains ready to print at all times and changes from a rest mode to an operative mode when a print request is received, and even while in operative mode, the bi-directional communication mode setting remains at intermittent bi-directional communication mode. More specifically, according to Tokura, even when the user has selected the intermittent bi-directional communication mode in which the bi-directional communication with the printer

Takayuki SHIMATANI, Application No. 10/578,232   Dkt. 2271/76217
Page 15

107 is not always performed, the bi-directional communication with the printer is established in quick response to the print request by shifting to the operative mode, and after completion of the printing, the bi-directional communication is shifted from the operative mode to the rest mode. Stated another way, the intermittent bi-directional communication mode setting does not change throughput such operative-to-rest-to-operative mode changes.

However, there is no disclosure or suggestion in Tokura of storing printer information and a two-way communication flag in a registry of the client computer.

In addition, Tokura is silent with respect to the "two-way flag of the printer information".

It is contended in the the Office Action that a two-way communication flag stored in a registry is *inherently* present in Tokura, or the actions based on operator selection in step (1) of figs. 2 and 3 will not be possible.

Applicant disagrees. In order for such contention in the Office Action to be valid, each and every implementation of the proposal of Tokura must require (i) a two-way communication flag that indicates whether two-way communication is enabled or disabled, and (ii) such flag is stored in a registry of the client computer.

Neither (i) nor (ii) is required, in each and every instance. For example, one skilled in the art would appreciate, from the teaching of Tokura, that two-way communication can be enabled throughout, and such communication is, under certain circumstances, performed under controlled timing.

Further, a flag, much less a flag in a registry, is not necessary for specifying whether two-way communication is enabled or disabled. Disabling of two-way communication can be effected in any of various techniques known in computer technologies that do not involve such a flag, such as interrupt processing, communication interface processing, etc.

Takayuki SHIMATANI, Application No. 10/578,232   Dkt. 2271/76217
Page 16

In short, Tokura, like Kadota (and Ohta), does NOT disclose or suggest the above-mentioned aspects (a) through (c) of the present application.

Applicant submits that the cited art, even when considered along with common sense and common knowledge to one skilled in the art, does *NOT* render unpatentable the above-mentioned aspects of the present application.

Accordingly, applicant respectfully submits that independent claims 1 and 3, and the claims depending therefrom, are allowable over the cited art.   Independent claim 6 and the claims depending therefrom are allowable over the cited art for similar reasons.

In view of the remarks hereinabove, applicant submits that the application is now allowable, and earnestly solicits the allowance of the application.

If a petition for an extension of time is required to make this response timely, this paper should be considered to be such a petition. The Patent Office is hereby authorized to charge any required fees in connection with this amendment, and to credit any overpayment, to our Deposit Account No. 03-3125.

If a telephone interview could advance the prosecution of this application, the Examiner is respectfully requested to call the undersigned attorney.

Respectfully submitted,

Paul Teng, Reg. No. 40,837
Attorney for Applicant
COOPER & DUNHAM LLP
Tel.: (212) 278-0400

EXHIBIT A

to

AMENDMENT

(U.S. Application No. 10/578,232)